Amendments to the Specification

Paragraph **[0216]** of the specification is amended as follows:

The combined control and data path portions of IEU 104 are shown in FIG. 5. The primary data path begins with the instruction/operand data bus 124 from the IFU 102. As a data bus, immediate operands are provided to an operand alignment unit 470 and passed on to a register file (REG ~~ARRAY~~ FILE) 472. Register data is provided from the register file 472 through a bypass unit 474, via a register file output bus 476, to a parallel array of functional computing elements (FU_{0-n}) 478_{0-n}, via a distribution bus 480. Data generated by the functional units 478_{0-n} is provided back to the bypass unit 474 or the register ~~array~~ file 472, or both, via an output bus 482.

Paragraph **[0217]** of the specification is amended as follows:

A load/store unit 484 completes the data path portion of the IEU 104. The load/store unit 484 is responsible for managing the transfer of data between the IEU 104 and CCU 106. Specifically, load data obtained from the data cache 134 of the CCU 106 is transferred by the load/store unit 484 to an input of the register ~~array~~ file 472 via a load data bus 486. Data to be stored to the data cache 134 of the CCU 106 is received from the functional unit distribution bus 480.

Paragraph **[0224]** of the specification is amended as follows:

The remaining units of the IEU control path include a retirement control unit 500, a control flow control (CF CTl) unit 528, and a done control (DONE CTL) unit 540. The retirement control unit 500 operates to void or confirm the execution of out-of-order executed instructions. Where an instruction has been executed out-of-order, that instruction can be confirmed or retired once all prior instructions have also been retired. Based on an identification of which of the current set of eight pending instructions have been executed provided on the control lines 532, the retirement control unit 500 provides control signals on control lines 534 coupled to the bus 518 to effectively confirm the result data stored by the register ~~array~~ file 472 as the result of the prior execution of an out-of-order executed instruction.

Paragraph **[0226]** of the specification is amended as follows:

The control flow control unit 528 performs the somewhat more specific function of detecting the logical branch result of each conditional branch instruction. The control flow control unit 528 receives an 8 bit vector identification of the currently pending conditional branch instructions from the EDecode unit 490 via the control lines 510. An 8 bit vector instruction done control signal is similarly received via the control lines 532 or 542 from the done control unit 540. This done control signal allows the control flow control unit 528 to identify when a conditional branch instruction is done at least to a point sufficient to determine a conditional control flow status. The control flow status

result for the pending conditional branch instructions are stored by the control flow control unit 528 as they are executed. The data necessary to determine the conditional control flow instruction outcome is obtained from temporary status registers in the register ~~array~~ file 472 via the control lines 530. As each conditional control flow instruction is executed, the control flow control unit provides a new control flow result signal on the control lines 348 to the IFU 102. This control flow result signal preferably includes two 8 bit vectors defining whether the status results, by respective bit position, of the eight potentially pending control flow instruction are known and the corresponding status result states, also given by bit position correspondence.

Paragraph **[0230]** of the specification is amended as follows:

The preferred generic architecture of a data path register file is shown in FIG. 6A. The data path register file 550 includes a temporary buffer 552, a register file array 554, an input selector 559, and an output selector 556. Data ultimately destined for the register file array 554 is typically first received by the temporary buffer 552 through a combined data input bus 558'. That is, all data directed to the data path register file 550 is multiplexed by the input selector 559 from a number of input buses 558, preferably two, onto the input bus 558'. Register select and enable control signals provided on the control bus 518 select the register location for the received data within the temporary buffer 552. On retirement of an instruction that produced data stored in the temporary buffer, control signals again provided on the control bus 518 enable the transfer of the data from the temporary buffer 552 to a logically corresponding register within the

register file array 554 via the data bus 560. However, prior to retirement of the instruction, data stored in the registers of the temporary buffer 552 may be utilized in the execution of subsequent instructions by routing the temporary buffer stored data to the output data selector 556 via a bypass portion of the data bus 560. The selector 556, controlled by a control signal provided via the control bus 518 selects between data provided from the registers of the temporary buffer 552 and of the register file array 554. The resulting data is provided on the register file output bus 563. Also, where an executing instruction will be retired on completion, i.e., the instruction has been executed in-order, the input selector 559 can be directed to route the result data directly to the register file array 554 via bypass extension 558".

Paragraph **[0241]** of the specification is amended as follows:

A number of additional special registers are at least logically present in the register ~~array~~ file 472. The registers that are physically present in the register ~~array~~ file 472, as shown in FIG. 7C, include a kernel stack pointer 568, processor state register (PSR) 569, previous processor state register (PPSR) 570, and an array of eight temporary processor state registers (tPSR[0..7]) 571. The remaining special registers are distributed throughout various parts of the architecture 100. The special address and data bus 354 is provided to select and transfer data between the special registers and the "A" and "B" sets of registers. A special register move instruction is provided to select a register from either the "A" or "B" register set, the direction of transfer and to specify the address identifier of a special register.

Paragraph **[0281]** of the specification is amended as follows:

The register rename unit 496 performs the additional function of selecting, via control signals provided on the bus 518 to the register file ~~array~~ 472, the source registers for access in the execution of the identified instructions. Destination registers for out-of-order executed instructions are selected as being in the temporary buffers 612, 680, 728 of the corresponding data path. In-order executed instructions are retired on completion with result data being stored through to the register ~~files~~ file arrays 614, 684, 732. The selection of source registers depends on whether the register has been prior selected as a destination and the corresponding prior instruction has not yet been retired. In such an instance, the source register is selected from the corresponding temporary buffer 612, 680, 728. Where the prior instruction has been retired, then the register of the corresponding register file array 614, 684, 732 is selected. Consequently, the register rename unit 496 operates to effectively substitute temporary buffer register references for register file register references in the case of out-of-order executed instructions.

Paragraph **[0284]** of the specification is amended as follows:

Upon identification of the instructions to issue, the register rename unit 496 initiates a register file access that continues to the end of the third processor cycle, P2. At the beginning of processor cycle P3, the instruction issuer unit 498 initiates operation by one or more of the. functional units 4780-n, such as shown as "Execute 0" , to receive and process source data provided from the register file ~~array~~ 472.

Paragraph **[0289]** of the specification is amended as follows:

The done control unit 540 monitors the functional units $478_{0\text{-}n}$ for the completion status of their current operations. In the preferred architecture 100, the done control unit 540 anticipates the completion of operations by each functional unit sufficient to provide a completion vector, reflecting the status of the execution of each instruction in the currently pending set of instructions, to the register rename unit 496, bypass control unit 520 and retirement control unit 500 approximately one half processor cycle prior to the execution completion of an instruction by a functional unit $478_{0\text{-}n}$. This allows the instruction issuer unit 498, via the register rename unit 496, to consider the instruction completing functional units as available resources for the next instruction issuing cycle. The bypass control unit 520 is allowed to prepare to bypass data output by the functional unit through the bypass unit 474. Finally, the retirement control unit 500 may operate to retire the corresponding instruction simultaneous with the transfer of data from the functional unit $478_{0\text{-}n}$ to the register file ~~array~~ 472.

Paragraph **[0290]** of the specification is amended as follows:

In addition to the instruction done vector provided from the done control unit 540, the retirement control unit 500 monitors the oldest instruction set output from the EDecode output 490. As each instruction in instruction stream order is marked done by the done control unit 540, the retirement control unit 500 directs, via control signals provided on control lines 534, the transfer of data from the temporary buffer slot to the

corresponding instruction specified register file register location within the register file ~~array~~ 472. The PC Inc/Size control signals are provided on the control lines 344 for each one or more instruction simultaneously retired. Up to four instructions may be retired per processor cycle. Whenever an entire instruction set has been retired, an IFIFO read control signal is provided on the control line 342 to advance the IFIFO 264.

Paragraph **[0294]** of the specification is amended as follows:

The instruction issuer unit 498 operates closely in conjunction with the bypass control unit 520 to control the routing of data between the register file ~~array~~ 472 and the functional units $478_{0\text{-}n}$. The bypass control unit 520 operates in conjunction with the register file access, output and store phases of operation shown in FIG. 14. During a register file access, the bypass control unit 520 may recognize, via control lines 522, an access of a destination register within the register file ~~array~~ 472 that is in the process of being written during the output phase of execution of an instruction. In this case, the bypass control unit 520 directs the selection of data provided on the functional unit output bus 482 to be bypassed back to the functional unit distribution bus 480. Control over the bypass unit 520 is provided by the instruction issuer unit 498 via control lines 532.

In the Drawings:

Please substitute the fourteen (14) drawing replacement sheets, submitted herewith, for the pending formal drawing sheets, to account for amendments made to FIGS. 5, 6A, 10, 11 and 12.